Chapter 1

A POWER-AWARE, SATELLITE-BASED PARALLEL SIGNAL PROCESSING SCHEME

Patrick M. Shriver, pshriver@lanl.gov, Los Alamos Nat'l. Lab.
Maya B. Gokhale, maya@lanl.gov, *Project Lead- Los Alamos Nat'l. Lab.*
Scott D. Briles, briles@lanl.gov, Los Alamos Nat'l. Lab.
Dong-In Kang, dkang@isi.edu, Information Sciences Institute
Michael Cai, dmc@lanl.gov, Los Alamos Nat'l. Lab.
Kevin McCabe, kmccabe@lanl.gov, Los Alamos Nat'l. Lab.
Stephen P. Crago, crago@isi.edu, Information Sciences Institute
Jinwoo Suh, jsuh@isi.edu, Information Sciences Institute

Abstract Satellite subsystem power budgets typically have strict margin allocations that limit the on-board processing capability of the spacecraft. Subsystems are assigned a fixed, maximum power allocation and are managed in an on/off manner according to available power and operations schedule. For a remote-sensing satellite, this limitation can result in poorer detection performance of interesting signal events as well as static instrument or data collection settings. Power-aware computation techniques can be utilized to increase the capability of on-board processing of science data and give the remote-sensing system a greater degree of flexibility.

We investigate a power-aware, signal processing scheme used to study signals from lightning events in the Earth's atmosphere. Detection and analysis of these lightning signals is complicated by the frequency dispersion experienced by the signal in the ionosphere as well as the interfering anthropogenic signals. We outline a method using multiprocessor architecture to run processing algorithms which have varying rates of power consumption. A 6 order magnitude spectrum of energy usage for these algorithms is obtained from experiment results.

Keywords: PAMA, satellite power management, power-aware remote sensing, ionospheric-dispersion signal, FORTÉ

1. INTRODUCTION

Conventional solutions to satellite power management consist of maintaining strict power budget margins during design and coarse on/off power switching of subsystems during operation. Power allocations are generally static and it is commonplace to assign a maximum power requirement to each subsystem. These methods can increase the time of the iterative design process (and thereby all associative costs of labor, overhead, etc.), be wasteful of the power resources, or require careful ground support planning of the science-instrument observation schedule, such as turning certain instruments off so that others may be turned on given the limited power available.

In this paper, we explore the capabilities of a power-aware, satellite-based computing system for on-board signal processing, to detect radio frequency (RF) events caused by natural events. The detection performance of the satellite-based remote sensing system can be improved on a moment-by-moment basis through the use of power-aware computing principles. Detection performance is directly related to how well and how often post-detection numerical computations can be executed to reduce false alarms. A greater capacity to reduce false alarms allows for a greater probability of detection. This capacity to reduce false alarms comes from the availability of computational resources, which, in turn, are dictated by power availability. This paper addresses an "intelligent" power-management technique that can be utilized in Department of Energy (DoE)/Department of Defense (DoD) as well as civil-satellite, remote-sensing applications, using the University of Southern California (USC)/Information Sciences Institute (ISI) Power Aware Multiprocessor Architecture (PAMA).

With the PAMA multiprocessors, power can be controlled in a gradient manner, in contrast to conventional techniques. This allows for more flexibility in the power budget margins and a higher degree of contingency options for the systems engineer. In our remote-sensing application, the multi-processor hardware, consisting of programmable processors and interconnect, is used to manage the data processing algorithms. The computational processing is adjusted to conserve or drain power according to the amount of power available vs. the rate of triggering events.

One remote-sensing application utilizing power-aware management techniques is in the processing of RF signals, e.g., lightning, in the Earth's atmosphere, similar to the mission of the DoE-funded Fast On-Orbit Recording of Transient Events (FORTÉ) satellite. FORTÉ was built by Los Alamos National Laboratory (LANL) and Sandia National

Laboratory (SNL); flight operations have also been shared as a joint venture between LANL and SNL. The principal goal of FORTÉ is to develop a comprehensive understanding of the correlation between the optical flash and very high frequency emissions from lightning. The signal processing techniques used in analyzing RF signals is the focus for this power-aware application.

The PAMA system can be used in a more sophisticated approach as part of an "intelligent" power-management scheme. Power resources are utilized and managed in a gradient manner, as opposed to the binary on/off operations or wasted as heat. In a remote-sensing application, the computational power of processing data can be adjusted to either conserve or exploit excess power as needed. This has the advantage of more on-board processing as power is available, resulting in a quicker ground-based analysis of the science data. Additionally, in a "smart" data processing scheme, it would also be possible to reorient or re-calibrate instruments based on the incoming data while in-orbit, without the delays associated with ground communications and analysis. Processing decisions are made based upon the available power, the health status of the satellite, and the rate of "interesting" science events. During intervals of low activity, or "off-peak," there can be low rates of data collection and processing; likewise, during "peak" periods, there can be high rates of data collection and processing. Thus, with these aspects, application-oriented power management can be a valuable tool for the spacecraft designer, allowing for greater flexibility in the payload power budget margins, more contingencies for handling power fluctuations during on-orbit operations, and a better management of science data collection and analysis.

It should be noted that our approach is a management technique, not a method to minimize subsystem power, i.e., power *management* instead of power *efficiency*. Over the traditional satellite power-management methods, this approach is a "smarter" algorithmic approach to power management. The advantage is that satellites which are aware of power usage and the overall satellite state can distribute power throughout subsystems to make the best use of the available power.

2. REMOTE-SENSING APPLICATION

For this work, we have focused on power-aware processing for a remote-sensing application similar in nature to the mission of FORTÉ. The FORTÉ satellite was launched in August of 1997 and carries a suite of instruments used for studying the optical and RF signals from lightning in the Earth's atmosphere. The results from FORTÉ have led to a

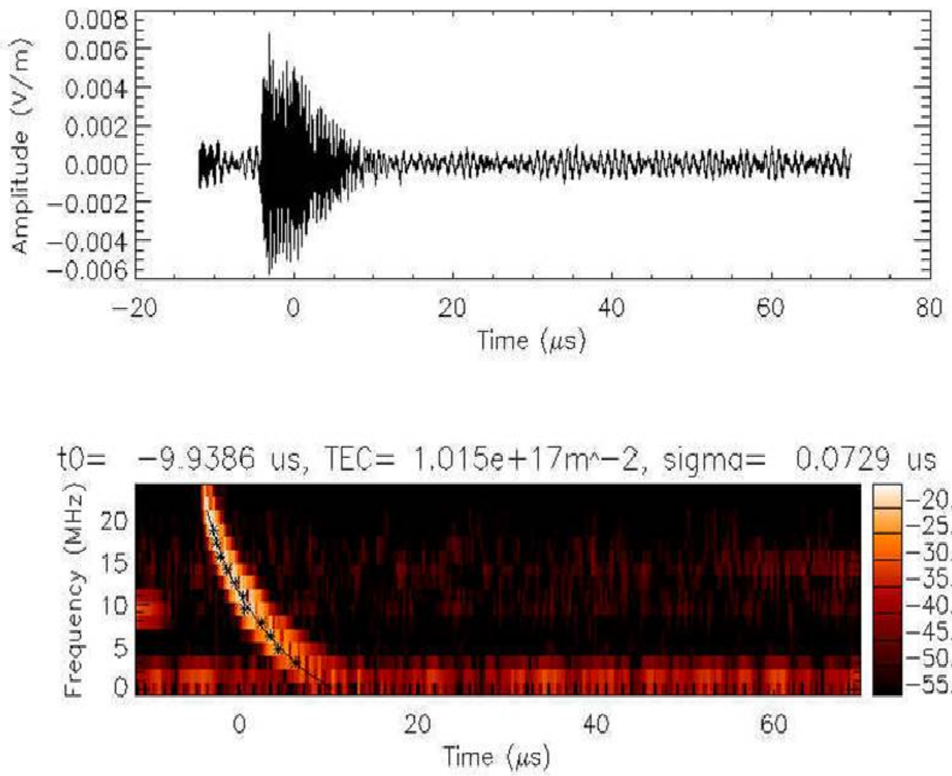

Figure 1.1. Ionic-Dispersed RF Signal

better understanding of the relationship between optical and RF lightning events, and future satellite missions can even use this knowledge to help provide global lightning and severe-storm monitoring [Russel-Dupré et al., 2001]. The processing algorithm for the RF lightning signals has been chosen for the multi-processor-based, power-aware application study.

2.1 Ionospheric-Dispersed Signals

A RF lightning event in the Earth's atmosphere generates a dispersed signal, i.e., low frequencies of the signal are delayed, as it propagates through the ionosphere. This is known as a "chirped" signal. A simulated chirped signal is shown by the graphs in Figure 1.1. The top graph is an illustration of the time-domain signal, and the bottom graph is a plot of the dispersed signal frequencies vs. the corresponding times.

The time taken for a given frequency of the chirped signal to arrive at the on-orbit receiver is related to the total electron content of the ionosphere along the direction of the signal travel, the given frequency, and the signal time-of-arrival if ionospheric dispersion did not exist [Enemark and Shipley, 1994]. This relationship is illustrated in the bottom graph of Figure 1.1 and can be determined from Eq. 1.1.

$$T_f = \frac{5.3x10^{-6} N_e}{4\pi^2} \frac{1}{f^2} + T_{oa} \tag{1.1}$$

where: T_f = frequency time-of-arrival
N_e = total electron content along the signal path
f = frequency
T_{oa} = signal time-of-arrival, neglecting ionospheric dispersion

The N_e, or total electron content (TEC), represents the number of electrons in a unit-area cross-section of an ionospheric column along the signal path. This atmospheric property is related to the propagation of radio signals through the ionosphere which can distort or bend the signals over the horizon. TEC is also related to the surface temperature of the Earth, and thus, could be viewed as an indicator for storm severity [NOAA, 1999].

The T_{oa} is the time the signal would have arrived at the on-orbit receiver if the ionosphere did not distort the signal; on the other hand, the first term on the right-hand side of Eq. 1.1 is the additional time taken due to the frequency dispersion. Notice that for higher frequencies, the time taken approaches that of T_{oa}, and for lower frequencies the time delay is greater. The T_{oa} parameter is primarily useful for geolocation, i.e., determining the geographic origin of the signal.

2.2 FORTÉ RF Hardware

FORTÉ receives RF signals either from two orthogonal monopoles mounted at the satellite's base or by passive moderate-gain antennas mounted on a 35 foot nadir-directed boom. There are two types of receivers tunable in a 30-300 MHz band which consist of a mixer, bandpass filter, and a second mixer stage. The first mixer up-converts the antenna signal to a higher frequency then passes the signal through the bandpass filter. The second mixer then converts the band-limited signal to baseband. Depending on the type of receiver, either a 12-bit high-speed digitizer or a 8-bit digitizer is used. The digitizers are in constant operation. An analog trigger box processes the output from the second-stage mixer and determines whether or not the digitized data is to be recorded in payload memory. The recorded data can then be downlinked to the SNL or University of Alaska Fairbanks groundstations. Data analysis is carried out as part of the ground operation at LANL and SNL.

The analog signal is passed into separate channels through a set of bandpass filters in the trigger box. The triggering signal, which determines the recording of data, is generated by predetermined threshold levels in each of these channels. Setting these threshold levels causes a trade-off relationship between the probabilities of true signal detection and probability of false alarms as illustrated in Figure 1.2. As the threshold levels are increased, fewer trigger signals are seen in each channel and the probability of false alarms decreases. However, this also decreases the number of detections. Thus, for better detection performance, more false alarms must be accepted. Once these threshold levels are set, the probability of detection cannot be improved, but post-detection techniques can shift the operating points of Figure 1.2 to the left, decreasing the rate of false alarms for a given rate of detection. Therefore, it is desirable to maintain the optimum value of detections vs. false alarms, a task normally accomplished by ground staff. The RF environment is very dynamic due to anthropogenic signals. In using a method which allows more on-board processing, the remote-sensing system can adjust more quickly to incoming signals than the ground-based approach.

3. SIGNAL FILTERS FOR PARAMETER ESTIMATION

The power-management activities in this remote-sensing application study are threefold:

- Increase the amount of on-board signal processing to reduce the probability of false alarms without affecting detection performance.
- Adjust the signal sample rate or signal sample capability.
- Control the algorithm power usage by varying the clock frequency, number of active processors, and active software modules.

These activities are constrained by the power state of the satellite and by the rate of incoming events.

The objective of the signal processing algorithms (see [Oppenheim and Schafer, 1989], [Press et al., 1992], [Kay, 1993] for further background on signal processing algorithms) is to reduce the number of false alarms and

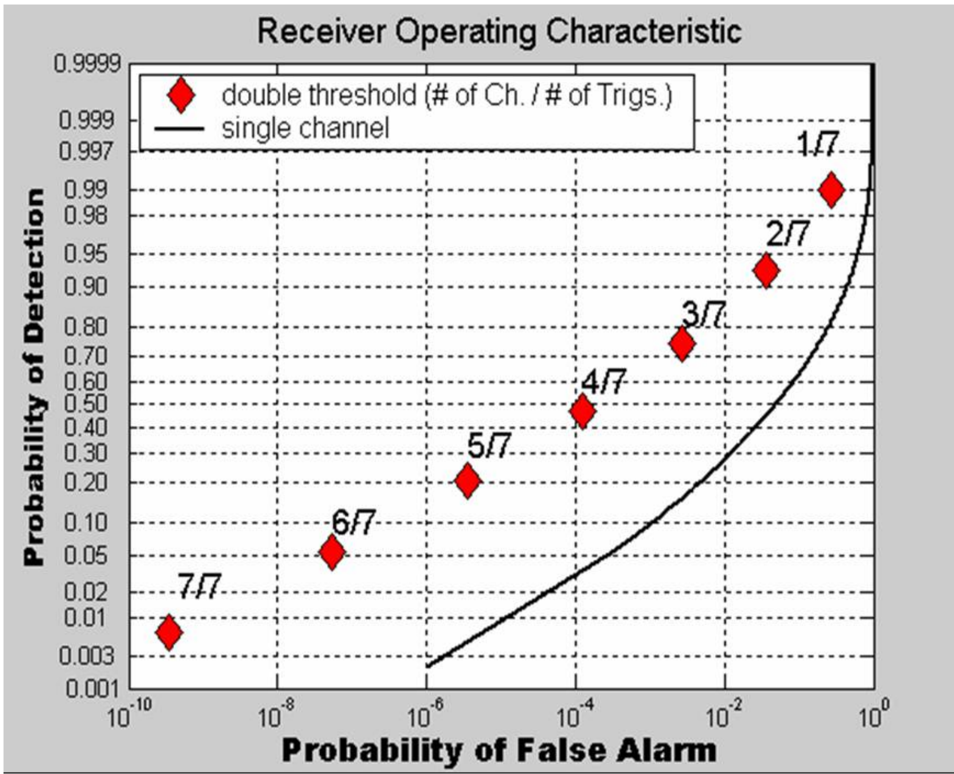

Figure 1.2. Signal Detection and False Alarm Rate Operating Curve

estimate the N_e and T_{oa} parameters, as described by Eq. 1.1, from the chirped signals received in orbit. The parameter-estimation flowchart is depicted in Figure 1.3. Note that for FORTÉ, the trigger and digitizer output signals are normally downlinked directly to the ground for analysis; there is no on-board processing of this data.

3.1 Trigger and Digitizer Output Signals

The time-domain, analog signal, $s(t)$, is to be passed through a hardware trigger box and a digitizer. The output from the trigger box and the digitizer provides two input data sets to the signal processing algorithm software. In hardware, the processing software is not active until a signal exceeds a given number of predetermined threshold levels, i.e., an event is not cataloged until "N" many of "M" channels pass threshold, known as a "big TRIGGER."

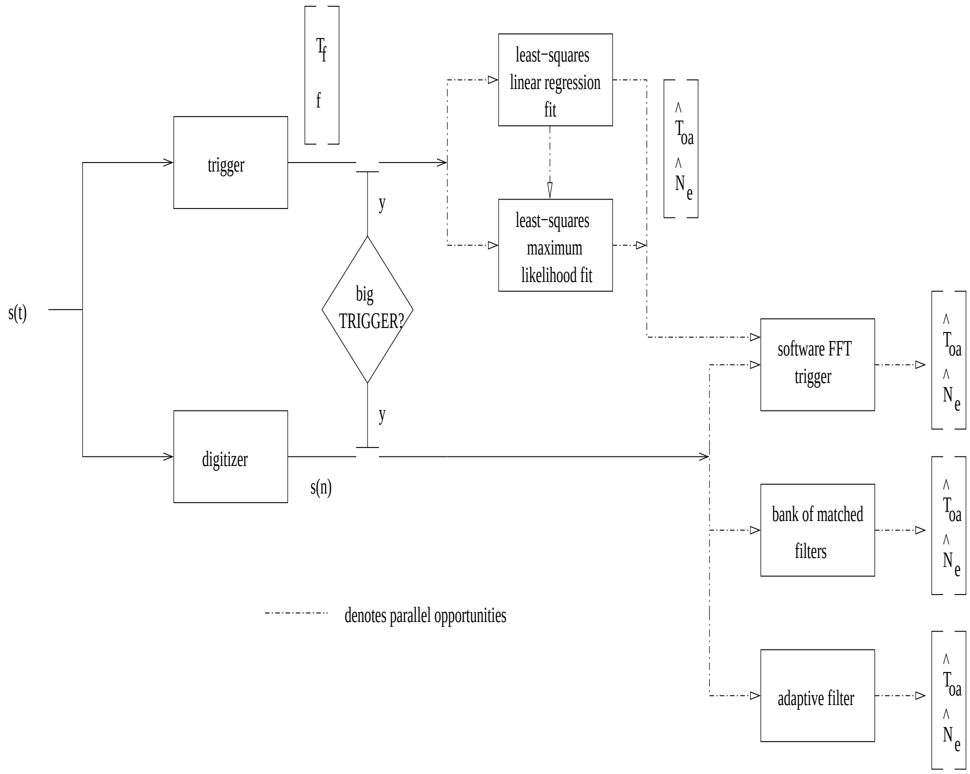

Figure 1.3. Parameter-Estimation, Algorithm Flowchart

The output from the trigger box consists of the f and T_f values obtained from filtering the signal through a set of bandpass filters, square law, and low-pass filters. Hence, f and T_f are the signal *center* frequency and *center* frequency time-of-arrival, respectively, as obtained by the filters. This data set will be passed into two routines that will perform a least-squares linear regression fit and a least-squares maximum likelihood fit to the data. These two routines will determine the initial N_e and T_{oa} estimates, $\hat{N}_e$ and $\hat{T}_{oa}$. The maximum likelihood fit is used to help reduce the number of false-alarm outliers. Both routines are deterministic code, i.e., the routines execute in a known finite amount of time.

3.1.1 Signal Filtering. The digitized signal data set $s(n)$ can be passed through the following set of filters to perform more refined estimates of $\hat{N}_e$ and $\hat{T}_{oa}$:

i. software FFT trigger This routine performs a Fast-Fourier Transform (FFT) of the signal and requires the initial estimates of $\hat{N}_e$ and $\hat{T}_{oa}$ from the least-squares modules.

ii. bank of matched filters The number of matched filters can be variable dependent upon the time allowed to process the current event before another event arrives, and the available computing power. This routine will take more time to execute than either the least-squares or the software FFT trigger routines.

iii. adaptive filter This filter routine will follow the adaptive least-mean-squares filter algorithm and use either a fixed or variable step size. The execution time for this routine is indeterminate but depends upon the quality of the signal and initial starting conditions. This filter should yield the most accurate estimates however.

Ideally, in a *conventional* computational satellite system, given enough computing power and time to process the signal, the routines of Figure 1.3 would be executed sequentially; however, given a more realistic scenario in which there is only a finite amount of time before the next event arrives and a finite amount of computing power in which to process the signal, these activities are currently reserved for ground-based analyses. Power-aware management principles can be utilized to process the data through these routines on-board the spacecraft. These ideas are realized through the PAMA computer system by managing the number of active processors working in parallel and customizing the interconnect to the desired communications paths.

4. ADAPTIVE POWER-AWARE PARALLEL PROCESSING

Our goal is to determine as accurately as possible an estimate for total electron content and event time of arrival given a varying power budget and varying inter-event duration. As discussed previously in Section 3.1.1, there is a suite of successively more powerful filters that can be applied. We have available multiple nodes of the PAMA parallel processor to apply in parallel to the signal processing tasks.

4.1 PAMA System Architecture

PAMA consists of a 4-node multi-processor connected by a programmable interconnect. On the PAMA-2 board, each node is a Hitachi SH-4 processor, with 32-bit integer and floating point hardware. Systems software includes the Linux operating system, MPI-like communications between processes, and a power-aware software library that allows the application to query power levels and set processor mode, clock frequency, and voltage.

4.2 Application Partitioning

Our application designates a distinguished node 0 as the application controller. Node 0 is responsible for

- obtaining the trigger data and data samples
- determining available power
- estimating the event rate
- distributing work to the three other nodes 1–3.

Worker nodes 1–3 all run the same program, consisting of the algorithms illustrated in Figure 1.3. Each worker receives trigger data and data samples over the interconnection network from Node 0. In addition, each worker receives a control vector from Node 0 telling it which of the filters to apply to the data. Although the workers run the same program, they usually take different paths through the program. Thus we refer to the worker nodes as operating in multiple program rather than single program mode.

We distinguish two types of operation. In the first type, multiple program, multiple data-stream processing, each worker node is given a set of filters to apply to a unique data stream. This mode is used when the event rate is high. The available power determines which filters a worker node will apply. In a low-power, high-event-rate scenario, the worker nodes will just perform the least means square fit (LMS) and possibly the maximum likelihood fit (ML). These routines are very quick (see Section 5 for quantitative results) and only require the trigger box data rather than the entire data sample stream. When more power is available, the controller may choose to have one or more of the worker nodes also compute a more time consuming filter on the data such as software trigger (ST), matched filter (MF), or adaptive filter (AF). Figure 1.4 illustrates a scenario in which Node 1 is asked to perform the complete parameter estimation suite, Node 2 runs the first four modules only, and Node 3 runs the first three modules only.

The second type of parallel processing is multiple program, single data stream mode. This mode is used in a high-power availability, low-event-rate scenario. The controller broadcasts the trigger and sample data to all workers, and, via the control vector, directs each worker to

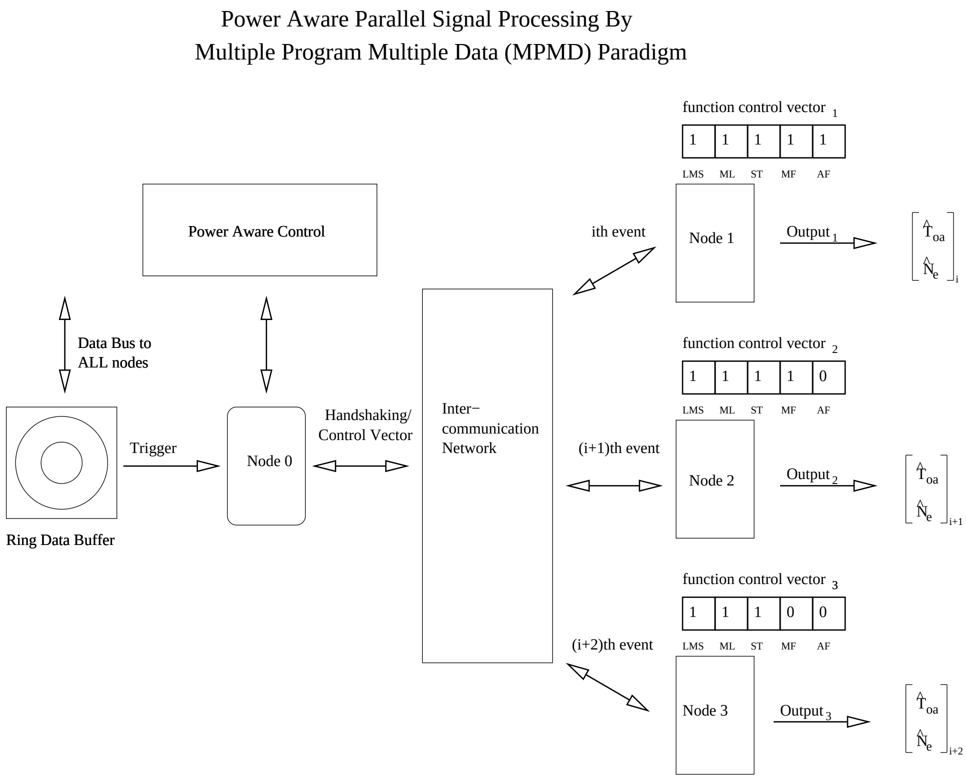

Figure 1.4. Multiple Program, Multiple Data Stream Parallel Processing

perform a different filter. In the case of another event occurring before all the workers are done, the controller may take the most accurate result computed to date and then reset the worker nodes to a new event. In the example of Figure 1.5, Node 1 performs the LMS, ML, and ST routines. Node 2 performs the LMS, ML, and MF; and Node 3 performs LMS, ML, and AF.

5. POWER AVAILABILITY AND USAGE

The control node must decide what subset of the signal filters each worker node should perform on what data. As mentioned above, this decision is based on power availability and the amount of power required by each of the signal filters.

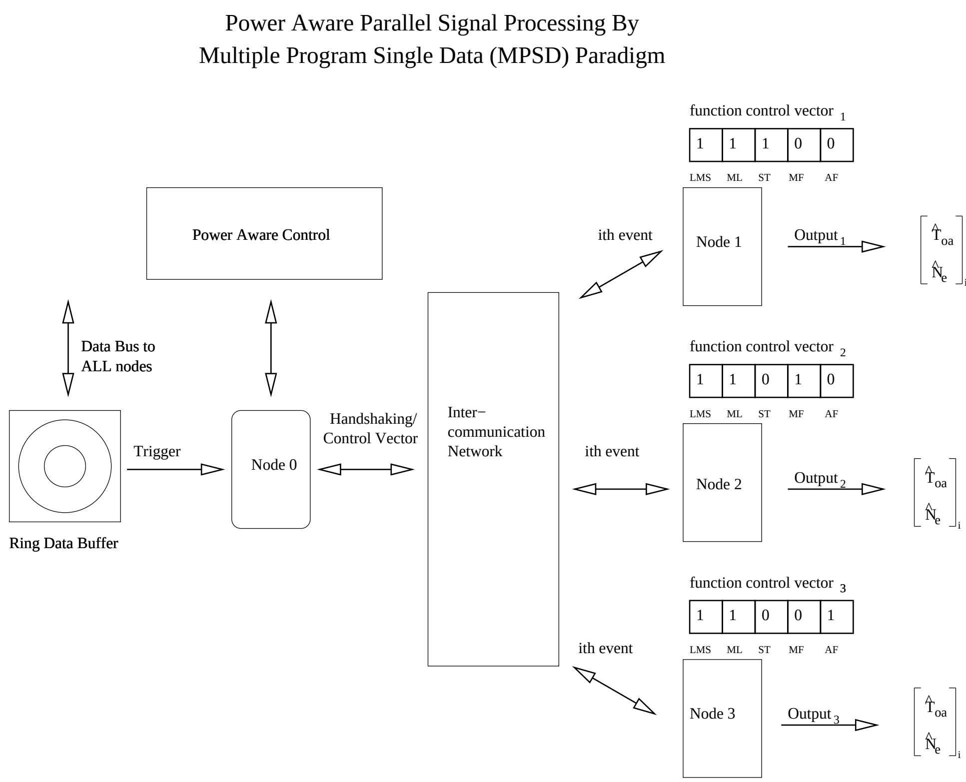

Figure 1.5. Multiple Program, Single Data Stream Parallel Processing

5.1 Power Availability

The amount of power available for signal processing computation is a complex function of orbit and housekeeping workload. Typically for solar array/battery satellites, the solar arrays will both recharge the batteries and provide power during "light" times. When in the "dark," i.e., eclipse, power is taken from the batteries. FORTÉ has the ability to draw from the solar arrays or battery or a combination of both. This is accomplished automatically in the hardware depending on the load. There are a few issues to consider for estimating the amount of future power available. These include the following:

- orbit eclipse times. When the satellite is in eclipse, the arrays are not charging the batteries, and no power is drawn directly from the solar arrays, hence, the batteries are the only source of power. There is also a limitation on how much power can be drawn from the batteries, typically known as the Depth of Discharge; beyond this limit, the batteries can no longer be recharged.

  Methods for estimating the amount of time spent in eclipse consist of using orbit propagator tools. For FORTÉ, eclipse times are predicted manually by using the Satellite Tool Kit program, commercially available from Analytical Graphics, Inc. The future load on the batteries and required recharging rates is then estimated based on these predictions. A similar method can be used for PAMA by uplinking the expected eclipse times. These times can then be loaded into the power-control algorithm.

- solar array degradation due to radiation. Typically a solar array will produce more power at the beginning-of-life than at end-of-life. This is dependent on the solar array properties and the radiation exposure for the given orbit. Degradation tends to be a long-term effect and can thus be monitored from the satellite health power-state values.

- sun incidence angle. The solar array will produce max power when the array is normal to the sun vector. Attitude data relative to the sun is needed for this information. For FORTÉ, the spacecraft body is covered with solar panels to provide a constant average power generated by the panels.

- rate of future events. The event rate can vary from several events a second to an event every fifteen minutes. For our science application, the event rate is determined principally by the satellite's orbit and time of year. In general, we expect more thunderstorm

> activity over land and in the summer than over water during colder seasons, thus leading to a higher event rate for the former. We have developed a simple model for the event rate that factors in these considerations. In operation, we will load the event rate as a table to the on-board processor.

For further information on satellite power system requirements, design, and estimation see [Larson and Wertz, 1992].

5.2 Power Usage

We have obtained power usage for four of the signal filters by running the filters on a variety of microprocessors. These include the following: a 733-MHz Pentium III processor running the Linux operating system and compiled with the GNU C compiler using optimization option O; a 200 MHz Hitachi SH-4 processor running the Linux operating system and compiled with the GNU C compiler using optimization option O; a Texas Instruments' TMS320C6711 with a clock frequency of 150 MHz, using no operating system and compiled using Code Composer Ver. 2.0 with the o3 optimization options; and a 266 MHz Power PC 750 running the VxWorks operating system. Time-to-execute values for each processor and for each signal-processing operation were determined. In addition, both time-to-execute values and power usage estimates (RMS and peak current) were determined for the Power PC 750.

Time-to-execute values that are presented in this paper are average values. A test set comprised of 21 test events was used for this benchmarking exercise. Each test event has hardware-trigger-box data and digitized waveform data associated with it. The data of the test set was synthetically generated using a MATLAB program that simulated a pulse event being received by a space-base receiver system containing a hardware trigger box and a waveform digitizer. Several (20 to 100) executions of the complete data set were performed to yield averaged time-to-execute values.

Time-to-execute values were determined by embedding compiler-specific timing functions into the C-language code. The timing functions involved the starting, stopping and/or reading of timers or clock-cycle counters. The method of timing was determined by what compiler-specific functions were available and familiar to the experimenter. Whatever the functions used, the placement was done in a manner as to only take into account the data manipulation only operations and not those operations associated with allocation of memory, movement of data or other operations that maybe dealt with differently in an application-specific, embedded-processors system.

Table 1.1. Timing of Signal Filters

	LMS	ML	ST	MF
Pentium/733	0.71μs	24.1μs	1.29ms	43.4ms
TI-C6711/150	4.6μs	112μs	14ms	1000ms
SH-4/200	7.14μs	152μs	11.6ms	516ms
PPC750/266	3.4μs	183μs	8.34ms	470ms

Table 1.2. Power Measurements for the PPC750

	LMS	ML	ST	MF
Current (amps-peak)	2.06	2.06	2.18	2.04
Power (Watts rms)	5.5	5.596	5.67	5.0
Execution Time	3.4μs	183μs	8.34ms	470ms
Energy (Joules)	18.7e-6	1.02e-3	47.3e-3	2.35

None of the development systems used to benchmark the code were developed to be application-specific system for the application presented in this paper. This being the case, the values presented in the Table 1.1 cannot be considered as inflexible values that can be used to determine design limits. However, the values do give a good presentation of the relative comparison of how the processors perform when tasked with the specific application of this paper. When examining the table, the clock frequency of the processor should be taken into account. Clock frequency has a directed impact on power usage, thus Tables 1.1 and 1.2 just begin to visit some of the trade space in the design of an application-specific system.

Power usage for the Power PC 750 executing the benchmarking code is presented in Table 1.2. The Jet Propulsion Laboratory (JPL) power-aware testbed consists of a Wind River PPC750 266MHz processor board that is running VxWorks 5.4.2. The processor operates at a constant 2.67V and current consumption is measured with a Tektronix TDS 7104 Digital Phosphor Oscilloscope. Current is sampled with the Tektronix TCP202 probe that is wired to the board. Software compilation is done with the VxWorks Tornado 2.0.2 programming tools which uses the GNU C compiler.

The software is compiled and downloaded to the testbed manually with the Tornado target server shell. The programs are run until an

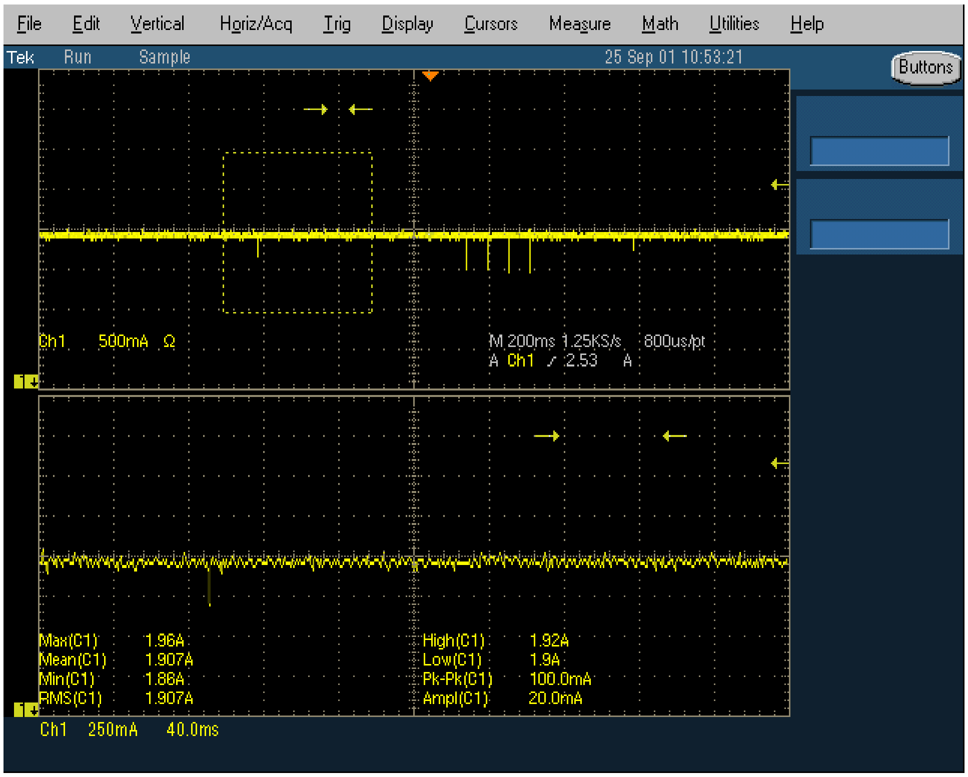

Figure 1.6. Matched Filter Background Measurements
(oscilloscope snapshot taken with JPL testbench equipment)

"average" current signal snapshot is taken with the oscilloscope. The "average" signal is determined manually by watching the current response during several program runs. The snapshot is taken when the current response produces a fairly consistent signal and consistent measured values. For reference, a "background" snapshot, i.e., when there is no user-program load, is also taken for each program test. Timing information is obtained using the "tickGet()" function in the software code. Since the ticks have a low resolution (in comparison to the program execution time) of 60 ticks/sec, an average value was calculated by summing the ticks over a significant number of program runs. Representative oscilloscope measurements for the matched filter are shown in Figures 1.6 and 1.7. Figure 1.6 shows the background measurement, and Figure 1.7 shows measurement while the matched filter routine is running.

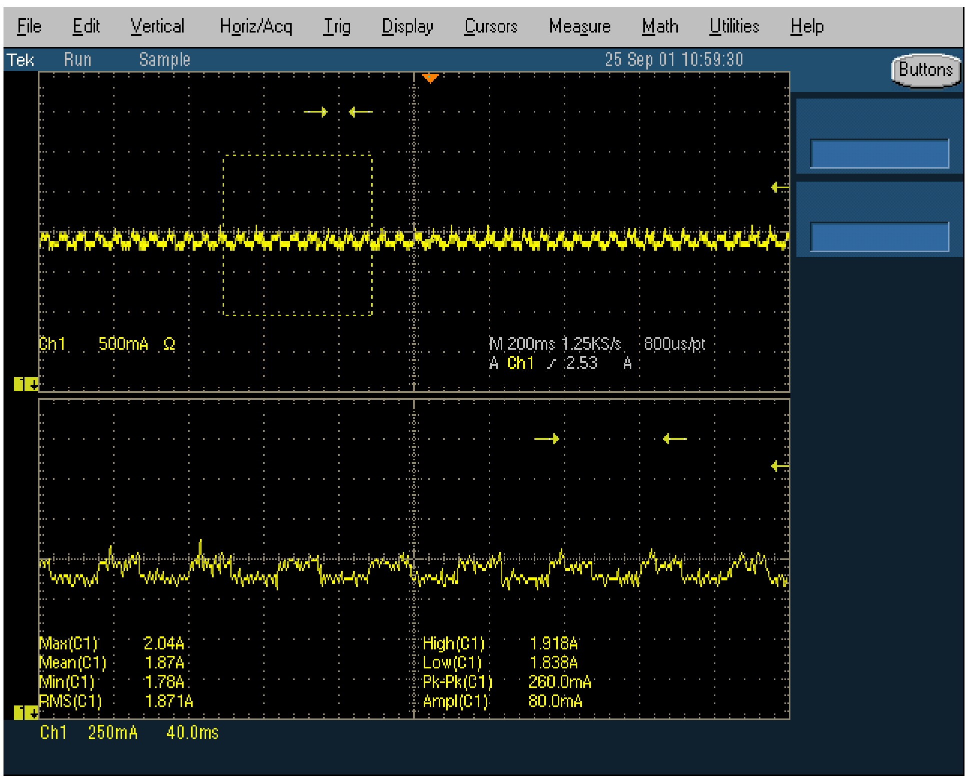

Figure 1.7. Matched Filter Run-time Measurements
(oscilloscope snapshot taken with JPL testbench equipment)

Both peak and average current were measured for a 2.67-volt power supply, supplying the processor. Table 1.2 provides the peak current, power (RMS), time-to-execute and energy-expended values for the four signal-processing operations. Since the application of this paper is satellite-based, it is believed that these values associated with power measurements would be the ones most desirable to a designer.

The test results of Tables 1.1 and 1.2 show a diverse spread of magnitudes from running the four signal processing routines LMS to MF. As expected, the LMS and ML routines execute in the least amount of time with the least amount of energy. These two routines are ideal for obtaining a quick, initial estimate of the parameters, $\hat{N}_e$ and $\hat{T}_{oa}$. The ST routine provides a more refined estimate of these parameters but also has an associated increase in the timing and energy costs. In terms of these performance metrics, the most expensive routine is the MF; it is approximately 3 orders of magnitude more costly in time than the LMS and ML routines, 6 orders of magnitude more costly than LMS in energy, and 3 orders of magnitude greater than ML in expended energy. Thus, these results illustrate the trade-off relationship between timing/energy costs and increasing parameter estimation accuracy: for higher-confidence estimates, there are more associative costs.

This parameter estimation trade space defines the operations of the power control node. It is desirable to obtain the best estimate possible given the constraints on both current and future available power, expected rates of incoming trigger events, and the timing/energy costs associated with routine execution.

Of worth noting is our exclusion of the StrongARM processor in our presentation of data. The reason for this is the belief that the processor would not be suitable for intense floating-point operation. This belief was supported by preliminary testing. As an example, a 206 MHz StrongARM processor required 443μs to perform the LMS fit and 30.39s to perform the MF operation. Although the StrongARM processor has many desirable features in terms of power-usage control, it was clear that it is unsuitable for this application.

6. CONCLUSIONS

In this paper we discuss how power-aware management techniques can be used in satellite remote-sensing applications as a "smarter" approach to satellite power management. A remote-sensing application, similar in nature to the DoE-funded FORTÈ satellite mission, and the signal processing algorithm used is described. Using a power-aware multi-processor, we have the ability of on-board post-processing of science data to reduce the rate of false alarms and give the remote-sensing system a greater degree of flexibility. We have described how the signal processing modules are mapped onto PAMA, exploiting multiple processors for more accurate detection when power is available, while scaling back to uni-processor and lesser clock-frequency mode when power is scarce. Experimental tests on four of the signal processing algorithms have yielded a 6 order of magnitude spectrum in energy consumption between the routines. As differing algorithms produce higher-confidence parameter estimates, significant increases in energy costs are experienced. This project is currently on-going and is a collaboration between LANL and USC/ISI.

Acknowledgments

This effort is sponsored by Defense Advanced Research Projects Agency (DARPA) through the Air Force Research Laboratory, USAF, under agreement number F30602-00-2-0548. We are indebted to JPL for the use of their Power PC testbench to measure algorithm power usage, and thanks specifically to Jeffrey Namkung of JPL for his invaluable assistance with the testing. Thanks are also due to the LANL FORTÉ team for supplying details of FORTÉ operation.

References

[Enemark and Shipley, 1994] Enemark, D. C. and Shipley, M. E. (1994). The FORTÉ receiver and sub-band triggering unit. In *8th Annual AIAA/Utah State University Conference on Small Satellites*.

[Kay, 1993] Kay, S. M. (1993). Fundamentals of statistical signal processing: Estimation theory.

[Larson and Wertz, 1992] Larson, W. J. and Wertz, J. R. (1992). *Space Mission Analysis and Design*, 2nd edition. pages 316–319 and 391–409.

[NOAA, 1999] NOAA (1999). http://www.ngs.noaa.gov/GRD/GPS/Projects/TEC.

[Oppenheim and Schafer, 1989] Oppenheim, A. and Schafer, R. (1989). *Discrete-Time Signal Processing.* pages 713–730.

[Press et al., 1992] Press, W. et al. (1992). *Numerical Recipes in C*, 2nd edition. pages 699–705.

[Russel-Dupré et al., 2001] Russel-Dupré, D., Klingner, P., et al. (2001). Four years of operations and results with FORTÉ. *American Institute of Aeronautics and Astronautics.*